

# PROGRAMMABLE CONTROLLER

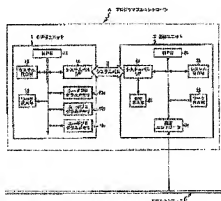
Patent number: JP8339210 (A)  
 Publication date: 1996-12-24  
 Inventor(s): YOSHIOKA MASATO +  
 Applicant(s): OMRON TATEISI ELECTRONICS CO +  
 Classification:  
 - international: G05B19/05; G06F13/00; G05B19/05; G06F13/00; (IPC1-7) G05B19/05; G06F13/00  
 - european:  
 Application number: JP19950147134 19950614  
 Priority number(s): JP19950147134 19950614

Also published as:

JP3358045 (B2)

## Abstract of JP 8339210 (A)

**PURPOSE:** To make execution of a user program follow in real time the change of the contact information that is shared with other programmable controllers by the communication carried out via a network. **CONSTITUTION:** A CPU unit 1 divides a user program into a shared contact user program and an independent contact user program and stores the former program in the memory 15b or 15c and the latter program in a memory 15a respectively. An MPU 11 usually carries out only the independent contact user program stored in the memory 15a. When the shared contact information is received via a network N, the MPU 11 interrupts the execution of the independent contact user program by interruption and carries out the shared contact user program stored in the memory 15b or 15c based on the shared contact information.



Data supplied from the **espacenet** database — Worldwide

特開平8-339210

(43) 公開日 平成8年(1996)12月24日

(51) Int.Cl. <sup>6</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 5 B 19/05			G 0 5 B 19/05	F
G 0 6 F 13/00	3 5 5	7368-5E	G 0 6 F 13/00	3 5 5

審査請求 未請求 請求項の数 5 O L (全 14 頁)

(21) 出願番号 特願平7-147134

(22) 出願日 平成7年(1995)6月14日

(71) 出願人 000002945

オムロン株式会社

京都府京都市右京区花園土堂町10番地

(72) 発明者 吉岡 誠人

京都府京都市右京区花園土堂町10番地 オムロン株式会社内

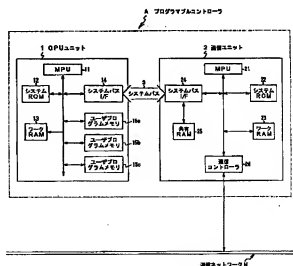
(74) 代理人 弁理士 和田 成則

## (54) 【発明の名称】 プログラマブルコントローラ

## (57) 【要約】

【目的】 ネットワーク介した通信によって他のプログラマブルコントローラと共有した共有接点情報の変化に、ユーザプログラム実行をリアルタイムに追従させる。

【構成】 CPUユニット1では、ユーザプログラムを共有接点ユーザプログラムと、独自接点ユーザプログラムとに分けて、独自接点ユーザプログラムはメモリ15aに格納し、共有接点ユーザプログラムはメモリ15b、15cに格納する。MPU11は、通常はメモリ15aに格納された独自接点ユーザプログラムのみを実行し、共有接点情報がネットワークNを介して送信されてきた場合には、割込みによってその独自接点ユーザプログラムの実行を中断して、その共有接点情報を使用してメモリ15aまたは15bに格納された共有接点ユーザプログラムを実行する。



## 【特許請求の範囲】

【請求項1】 ネットワークを介した通信によって他のプログラマブルコントローラと接点情報を共有し、その共有接点情報に基づきユーザプログラムを実行するプログラマブルコントローラにおいて、上記共有接点情報を使用する共有接点ユーザプログラムを格納した共有接点ユーザプログラムメモリと、上記共有接点情報を使用しない独自接点ユーザプログラムを格納した独自接点ユーザプログラムメモリと、通常は上記独自接点ユーザプログラムメモリに格納された独自接点ユーザプログラムを実行し、ネットワークを介し他のプログラマブルコントローラから共有接点情報が送信されてきた場合には、割込みにより上記独自接点ユーザプログラムの実行を中断し、上記共有接点ユーザプログラムメモリに格納された共有接点ユーザプログラムを上記送信されてきた共有接点情報を使用して実行するプログラム実行手段と、を具備することを特徴とするプログラマブルコントローラ。

【請求項2】 共有接点ユーザプログラムメモリには、複数の共有接点ユーザプログラムが格納され、かつ、共有接点情報には、当該共有接点情報を使用する共有接点ユーザプログラムを指示するプログラム指示情報が付加されており、プログラム実行手段は、ネットワークを介し他のプログラマブルコントローラから共有接点情報が送信されてきた場合には、割込みにより上記独自接点ユーザプログラムの実行を中断し、当該共有接点情報に付加されたプログラム指示情報に基づいて上記共有接点ユーザプログラムメモリに格納された対応する共有接点ユーザプログラムを讀出し、その共有接点ユーザプログラムを上記送信されてきた共有接点情報を使用して実行する、ことを特徴とする請求項1記載のプログラマブルコントローラ。

【請求項3】 複数の共有接点ユーザプログラムが各々使用する共有接点情報は、通信ポート毎に送受信され、かつ、その各共有接点情報にはプログラム指示情報として通信ポート番号が付加されるものとし、プログラム実行手段は、予め各通信ポート番号と、各通信ポートを介して送信されてくる各共有接点情報を使用する共有接点ユーザプログラムとを対応させた通信ポート情報テーブルを有しており、ネットワークを介し他のプログラマブルコントローラから共有接点情報が送信されてきた場合には、当該共有接点情報に付加されたプログラム指示情報としての通信ポート番号に基づき上記通信ポート情報テーブルを参照して共有接点ユーザプログラムメモリに格納された対応する共有接点ユーザプログラムを讀出す、ことを特徴とする請求項2記載のプログラマブルコント

ローラ。

【請求項4】 独自接点ユーザプログラムには、ネットワークを介して他のプログラマブルコントローラに対し共有接点情報の讀出しを要求する共有接点読出しコマンドを送信させる命令が含まれており、プログラム実行手段は、通常は上記独自接点ユーザプログラムメモリに格納された独自接点ユーザプログラムを実行し、上記命令を実行した場合には、ネットワークを介して他のプログラマブルコントローラに対し上記共有接点読出しコマンドを送信すると共に、ネットワークを介し当該他のプログラマブルコントローラから共有接点情報が送信されてきた場合には、割込みにより上記独自接点ユーザプログラムの実行を中断し、共有接点ユーザプログラムメモリに格納された共有接点ユーザプログラムを上記送信されてきた共有接点情報を使用して実行する、ことを特徴とする請求項1、請求項2または請求項3記載のプログラマブルコントローラ。

【請求項5】 ネットワークを介した通信によって他のプログラマブルコントローラと接点情報を共有し、その共有接点情報に基づきユーザプログラムを実行するプログラマブルコントローラにおいて、上記共有接点情報を使用しない独自接点ユーザプログラムを格納した独自接点ユーザプログラムメモリと、通常は上記独自接点ユーザプログラムメモリに格納された独自接点ユーザプログラムを実行し、ネットワークを介し他のプログラマブルコントローラから共有接点情報およびその共有接点情報を使用する共有接点ユーザプログラムが送信されてきた場合には、割込みにより上記独自接点ユーザプログラムの実行を中断し、上記送信されてきた共有接点情報を使用して上記共有接点ユーザプログラムを実行するプログラム実行手段と、を具備することを特徴とするプログラマブルコントローラ。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】本発明は、ネットワークを介した通信によって他のプログラマブルコントローラと接点情報を共有し、その共有接点情報に基づきユーザプログラムを実行するプログラマブルコントローラに関する。

## 【0002】

【従来の技術】近年、生産ライン等のスイッチやロボット等の各種FA機器の動作制御を行うプログラマブルコントローラの分野でも、ネットワークを介し他のプログラマブルコントローラと接続してネットワークシステムを構築することが行われており、このようなシステムでは、複数の各プログラマブルコントローラ間でネットワークを介した通信によって接点情報を共有することが行われている。

【0003】このようなプログラマブルコントローラで

は、従来、ネットワークを介し他のプログラマブルコントローラとの間で共有する接点情報（以下、「共有接点情報」という。）を使用する共有接点処理と、そのような共有接点情報が必要としない各プログラマブルコントローラ独自の接点情報（以下、「独自接点情報」という。）を使用する独自接点処理とを分けて混在させたユーザプログラムを、共有接点情報の通信とは無関係にサイクリックに実行していた。

【0004】

【発明が解決しようとする課題】しかし、このようなプログラマブルコントローラでは、共有接点情報を使用する共有接点処理と、ネットワークNを介した通信とは関係のない独自接点情報を使用する独自接点処理とが混在したユーザプログラムを、共有接点情報の通信とは無関係にサイクリックに実行していたため、他のプログラマブルコントローラからネットワークを介して更新された共有接点情報が送信されてきた場合でも、ユーザプログラムの実行中であれば、直ぐにその共有接点情報を使用できず、次サイクルまで待つ必要があるので、ユーザプログラムの実行が共有接点情報の変化にリアルタイムに追従できない、という問題があった。

【0005】特に、実行サイクルの長いユーザプログラムを実行している際に、共有接点情報がネットワークを介し送信されてきて入力した場合には、その入力タイミングからその共有接点情報を使用してユーザプログラムを実行するタイミングまで時間が分かる場合があり、共有接点情報を使用したリアルタイムな処理が実行できなかった。

【0006】本発明は、このような問題に着目してなされたもので、ユーザプログラムの実行がネットワークを介した共有接点情報の変化にリアルタイムに追従できるプログラマブルコントローラを提供することを目的とする。

【0007】

【課題を解決するための手段】上記目的を達成するため、請求項1記載の発明では、ネットワークを介した通信によって他のプログラマブルコントローラと接点情報を共有し、その共有接点情報に基づきユーザプログラムを実行するプログラマブルコントローラにおいて、上記共有接点情報を使用する共有接点ユーザプログラムを格納した共有接点ユーザプログラムメモリと、上記共有接点情報を使用しない独自接点ユーザプログラムを格納した独自接点ユーザプログラムメモリと、通常は上記独自接点ユーザプログラムメモリに格納された独自接点ユーザプログラムを実行し、ネットワークを介し他のプログラマブルコントローラから共有接点情報が送信されてきた場合には、割込みにより上記独自接点ユーザプログラムの実行を中断し、上記共有接点ユーザプログラムメモリに格納された共有接点ユーザプログラムを上記送信されてきた共有接点情報を使用して実行するプログラム実

行手段と、を具備することを特徴とする。

【0008】請求項2記載の発明では、請求項1記載のプログラマブルコントローラにおいて、共有接点ユーザプログラムメモリには、複数の共有接点ユーザプログラムが格納され、かつ、共有接点情報には、当該共有接点情報を使用する共有接点ユーザプログラムを指示するプログラム指示情報が付加されており、プログラム実行手段は、ネットワークを介し他のプログラマブルコントローラから共有接点情報が送信されてきた場合には、割込みにより上記独自接点ユーザプログラムの実行を中断し、当該共有接点情報に付加されたプログラム指示情報に基づいて上記共有接点ユーザプログラムメモリに格納された対応する共有接点ユーザプログラムを読出し、その共有接点ユーザプログラムを上記送信されてきた共有接点情報を使用して実行する、ことを特徴とする。

【0009】請求項3記載の発明では、請求項2記載のプログラマブルコントローラにおいて、複数の共有接点ユーザプログラムが各々使用する共有接点情報は、通信ポート毎に送受信され、かつ、その各共有接点情報にはプログラム指示情報として通信ポート番号が付加されるものとし、プログラム実行手段は、予め各通信ポート番号と、各通信ポートを介して送信されてくる各共有接点情報を使用する共有接点ユーザプログラムとを対応させた通信ポート情報テーブルを有しており、ネットワークを介し他のプログラマブルコントローラから共有接点情報が送信されてきた場合には、当該共有接点情報に付加されたプログラム指示情報としての通信ポート番号に基づき上記通信ポート情報テーブルを参照して共有接点ユーザプログラムメモリに格納された対応する共有接点ユーザプログラムを読出す、ことを特徴とする。

【0010】請求項4記載の発明では、請求項1、請求項2または請求項3記載のプログラマブルコントローラにおいて、独自接点ユーザプログラムには、ネットワークを介して他のプログラマブルコントローラに対し共有接点情報の読出しを要求する共有接点読出しコマンドを送信させる命令が含まれており、プログラム実行手段は、通常は上記独自接点ユーザプログラムメモリに格納された独自接点ユーザプログラムを実行し、上記命令を実行した場合には、ネットワークを介して他のプログラマブルコントローラに対し上記共有接点読出しコマンドを送信すると共に、ネットワークを介し当該他のプログラマブルコントローラから共有接点情報が送信されてきた場合には、割込みにより上記独自接点ユーザプログラムの実行を中断し、共有接点ユーザプログラムメモリに格納された共有接点ユーザプログラムを上記送信されてきた共有接点情報を使用して実行する、ことを特徴とする。

【0011】請求項5記載の発明では、ネットワークを介した通信によって他のプログラマブルコントローラと接点情報を共有し、その共有接点情報に基づきユーザ

プログラムを実行するプログラマブルコントローラにおいて、上記共有接点情報を使用しない独自接点ユーザプログラムを格納した独自接点ユーザプログラムメモリと、通常は上記独自接点ユーザプログラムメモリに格納された独自接点ユーザプログラムを実行し、ネットワークを介して他のプログラマブルコントローラから共有接点情報およびその共有接点情報を使用する共有接点ユーザプログラムが送信されてきた場合には、割込みにより上記独自接点ユーザプログラムの実行を中断し、上記送信されてきた共有接点情報を使用して上記共有接点ユーザプログラムを実行するプログラム実行手段と、を具備することを特徴とする。

【0012】

【作用】本発明では、共有接点情報を使用する共有接点ユーザプログラムと、共有接点情報を使用しない独自接点ユーザプログラムとを分けて格納しておき、通常は独自接点ユーザプログラムを実行し、ネットワークを介して他のプログラマブルコントローラから共有接点情報が送信されてきた場合には、割込みによりその独自接点ユーザプログラムの実行を中断し、送信されてきた共有接点情報を使用して共有接点ユーザプログラムを実行するようになる。

【0013】また、複数の共有接点ユーザプログラムをプログラム番号により区別し、共有接点情報には当該共有接点情報を使用する共有接点ユーザプログラムのプログラム番号を指示するプログラム指示情報を付加して送信するようにして、このような共有接点情報が送信がされてきた場合には、割込みにより独自接点ユーザプログラムの実行を中断し、当該共有接点情報に付加されたプログラム指示情報が指示する共有接点ユーザプログラムを、送信されてきた共有接点情報を使用して実行するようにしても良い。

【0014】さらに、本発明では、共有接点情報を使用する共有接点ユーザプログラムを所有せず、共有接点情報を使用しない独自接点ユーザプログラムのみを所有するようにして、通常は独自接点ユーザプログラムメモリに格納された独自接点ユーザプログラムを実行し、ネットワークを介して他のプログラマブルコントローラから共有接点情報およびその共有接点情報を使用する共有接点ユーザプログラムが送信されてきた場合には、割込みによりその独自接点ユーザプログラムの実行を中断し、その共有接点情報を使用して上記共有接点ユーザプログラムを実行するようにしても良い。

【0015】

【実施例】以下、本発明に係るプログラマブルコントローラの実施例を図面に基いて説明する。

【0016】図1に、本発明に係るプログラマブルコントローラの第1実施例の構成を示す。

【0017】このプログラマブルコントローラAは、図に示すように、CPUユニット1と、通信ユニット2と

から構成されており、内部的にはCPUユニット1と通信ユニット2とがシステムバス3を介して接続され、内部的には通信ユニット2を介して通信ネットワークNと接続されている。

【0018】CPUユニット1は、本ユニット1全体を制御するMPU11と、本ユニット1全体を制御するためのシステムプログラムが格納されたシステムROM12と、MPU11のワークエリアとして使用されるワークRAM13と、システムバスインタフェース（I/F）14と、各々異なるユーザプログラムが格納されたユーザプログラムメモリ15a～15cとを有している。

【0019】ここで、本実施例では、ユーザプログラムメモリ15aには、ネットワークNの通信を介した共有接点情報を使用しない独自接点情報を使用する独自接点ユーザプログラムが格納されており、ユーザプログラムメモリ15b、15cには、各々、ネットワークNの通信を介した共有接点情報を使用する共有接点ユーザプログラムが格納されており、共有接点ユーザプログラムが複数あるので、各々にプログラム番号が割り付けられているものとする。

【0020】通信ユニット2は、本ユニット2全体を制御するMPU21と、本ユニット2全体を制御するためのシステムプログラムが格納されたシステムROM22と、MPU21のワークエリアとして使用されるワークRAM23と、システムバスインタフェース（I/F）24と、システムインタフェース24に接続された共有RAM25と、ネットワークNに対しデータの送受信を行う通信コントローラ26とを有する。

【0021】共有RAM25は、通信ユニット2のMPU21だけでなく、CPUユニット1のMPU11がシステムバス3を介しアクセス可能で、MPU11がネットワークN上に送信すべき接点情報のデータやコマンド等を書き込んだり、あるいは通信ユニット2がネットワークN上から取り込んだ接点情報のデータやレスポンス等を読出するようにマルチポート、あるいはシングルポートでMPU11とMPU21とのアクセスをバスアービトレーションするように構成されている。

【0022】尚、この図では、図示していないが、この通信ネットワークN上には、以上のように構成された他のプログラマブルコントローラが複数台接続されており、ネットワークNを介した通信によりこのプログラマブルAとの間で共有接点情報を共有するように構成されている。

【0023】図2は、各ユーザプログラムメモリ15a～15cに格納される独自接点および共有接点ユーザプログラムのメモリマップを示している。

【0024】このメモリマップMに示すように、各ユーザプログラムメモリ15a～15cに格納されるユーザプログラムには、そのプログラム格納エリアの先頭アド

レスが割り当てられている。

【0025】具体的には、ユーザプログラムメモリ15aに格納される独自接点ユーザプログラムには、先頭アドレス1が割り当てられ、ユーザプログラムメモリ15bに格納される共有接点ユーザプログラム1には、先頭アドレス2が割り当てられ、ユーザプログラムメモリ15cに格納される共有接点ユーザプログラム2には、先頭アドレス3が割り当てられる。

【0026】尚、このメモリマップMでは、まだ他に共有接点ユーザプログラムがあれば、同様に先頭アドレスがnまで割り当てられることを示している。

【0027】図3は、CPUユニット11におけるユーザプログラム管理テーブルを示している。

【0028】このユーザプログラム管理テーブルT1は、プログラミングツール等によってユーザプログラムメモリ15a~15cに格納すべきユーザプログラムを作成する際に同時に作成されるもので、図2に示すメモリマップMを参照して、各ユーザプログラムの先頭アドレス1~nを格納して構成されており、MPU11内あるいはワークRAM13等に格納しておくものである。

【0029】次に、以上のように構成されたプログラマブルコントローラにおけるユーザプログラム実行処理を説明する。

【0030】図4に、このプログラマブルコントローラの全体の処理を示す。

【0031】このプログラマブルコントローラAでは、まず、MPU11が一般的な初期処理を行い(ステップ100)、続いてネットワークNの通信を介した共有接点情報を使用しない、すなわちこのプログラマブルコントローラA独自で他のプログラマブルコントローラとは関係のない独自接点情報を使用し、ユーザプログラムメモリ15aに格納された独自接点ユーザプログラムを実行する後述の独自接点処理を行い(ステップ200)、ネットワークNを介し他のプログラマブルコントローラから共有接点情報等を含む各種コマンドやレスポンスが送られてくるまで(ステップ300“No”)、この独自接点処理(ステップ200)を繰り返すようにする。

【0032】そして、この独自接点処理中にネットワークNを介し他のプログラマブルコントローラから共有接点情報を含む各種コマンドやレスポンスが送られてきた場合には(ステップ300“Yes”)、通信ユニット2の通信コントローラ26がそれを受信して、システムバスインタフェース24を介し共有RAM25に書き込み、通信ユニット2のMPU21等がシステムバス3を介してCPUユニット1に対し他のプログラマブルコントローラからコマンドやレスポンスの受信があったことを割り込みにより通知する(ステップ400)。

【0033】CPUユニット1では、MPU11がシステムバスインタフェース14を介しその割り込み通知を受

けて、上記独自接点処理をいったん中断し、システムバス3を介し通信ユニット2の共有RAM25にアクセスして、ネットワークNを介して他のプログラマブルコントローラから通信されてきたコマンドやレスポンスを共有RAM25からワークRAM13に書き込み、かつ、解析して、その受信したコマンドやレスポンスに共有接点情報が含まれているか否かを判断する(ステップ500)。

【0034】ここで、ネットワークNを介し他のプログラマブルコントローラから受信したコマンドやレスポンスに共有接点情報が含まれていない場合には(ステップ500“No”)、その割込み処理を終了して、上記ステップ200の独自接点処理を継続して実行する一方、その受信したコマンドやレスポンスに共有接点情報が含まれている場合には(ステップ500“Yes”)、その共有接点情報等を使用してユーザプログラムメモリ15b、15cに格納された共有接点ユーザプログラムを実行する後述の共有接点処理を行い(ステップ600)、その共有接点処理の終了後は上記ステップ200の独自接点処理に戻る。

【0035】次に、図4におけるステップ200の独自接点処理を図面を参照して詳細に説明する。

【0036】図5に、図4におけるステップ200の独自接点処理を示す。

【0037】この独自接点処理では、MPU11が、まず、ユーザプログラムメモリ15aに格納された独自接点を使用する独自接点ユーザプログラムを讀出して実行すると共に(ステップ210)、その独自接点ユーザプログラム実行中にネットワークNを介し他のプログラマブルコントローラへ共有接点讀出しコマンドの送信を指示するN\_READ命令を実行したか否かを判断する(ステップ220)。

【0038】ここで、N\_READ命令を実行したものと判断した場合には、その独自接点ユーザプログラムの1スキャン分実行終了後、周辺処理の1つの処理として共有接点讀出しコマンドをネットワークNを介し当該他のプログラマブルコントローラに送信し(ステップ230)、その共有接点讀出しコマンドの送信後、データの入出力処理等の他の周辺処理を行い(ステップ240)、その周辺処理の終了後、独自接点ユーザプログラムの実行処理(ステップ210)に戻って以上の処理をサイクル的に繰り返すようにする。

【0039】図6に、図5におけるステップ230の共有接点讀出しコマンドの送信処理を詳細に示す。

【0040】この第1実施例では、ネットワークNを介し他のプログラマブルコントローラへの共有接点讀出しコマンドの送信を、上述の通り独自接点ユーザプログラム内に記述されたN\_READ命令の実行により行うようにしている。

【0041】また、本実施例では、共有接点情報はプロ

グラマブルコントローラAのチャネル単位で通信するため、このN\_READ命令には、この命令の送信相手となる相手プログラマブルコントローラのネットワークアドレスや、通信ポート番号（複数のプログラマブルコントローラに同時に共有接点読出しコマンドを送信する場合には、そのレスポンスを区別するための識別子となる。）や、共有接点読出しレスポンスを受信したときに起動する共有接点ユーザプログラムのプログラム番号、相手プログラマブルコントローラにおける共有接点情報の開始チャネル、相手プログラマブルコントローラから受信した共有接点情報等を格納するエリアの開始チャネル、および受信を要求するチャネル数等のパラメータを記述するようにする。

【0042】送信処理の動作を説明すると、MPU11は、N\_READ命令の実行により、まず、そのN\_READ命令の上記パラメータから相手プログラマブルコントローラのネットワークアドレスを取り出し、送信すべき共有接点読出しコマンドにセットすると共に（ステップ231）、自プログラマブルコントローラのネットワークアドレスも共有接点読出しコマンドにセットし（ステップ232）、さらにコマンドコードをセットする（ステップ233）。

【0043】次に、そのパラメータから通信ポート番号を取り出し、送信すべき共有接点読出しコマンドにセットすると共に（ステップ234）、そのパラメータから受信しようとする相手プログラマブルコントローラにおける共有接点情報の開始チャネルを取り出して共有接点読出しコマンドにセットし（ステップ235）、さらにそのパラメータに記述された受信要求チャネル数をその共有接点読出しコマンドにセットして（ステップ236）、指定ポートを用い上記のようにN\_READ命令の各種パラメータがセットされた共有接点読出しコマンドを送信する（ステップ237）。

【0044】その後、N\_READ命令のパラメータから当該共有接点読出しコマンドに対するレスポンスを受信したときに起動する共有接点ユーザプログラムのプログラム番号を取り出し、そのプログラム番号を図2に示すメモリマップM、および図3に示すユーザプログラムの管理テーブルT1を参照して各ユーザプログラムの先頭アドレスに変換して、後述する図8に示す通信ポート情報テーブルT2にセットすると共に（ステップ238）、そのパラメータから相手プログラマブルコントローラの接点情報格納エリア開始チャネルを取り出しその通信ポート情報テーブルT2にセットして（ステップ239）、このN\_READ命令の実行による共有接点読出しコマンド送信処理が終了する。

【0045】図7に、図6に示すN\_READ命令の処理によって送信される共有接点読出しコマンドのフォーマットを示す。

【0046】この共有接点読出しコマンドC1は、上記

図6に示す処理により、相手プログラマブルコントローラのネットワークアドレス、自プログラマブルコントローラのネットワークアドレス、コマンド/レスポンスコード、通信ポート番号、受信しようとする相手プログラマブルコントローラからの共有接点情報の開始チャネル、および受信を要求するチャネル数がセットされて送信されることになる。

【0047】図8は、図6に示すステップ238、239の処理で作成される通信ポート情報テーブルT2を示している。

【0048】この通信ポート情報テーブルT2には、図に示すように、通信ポート毎にそのポート番号0～m（mは自然数）に対応させて、共有接点読出しコマンドを送信した相手プログラマブルコントローラからの共有接点読出しレスポンスを受信した際に起動する共有接点ユーザプログラムの先頭アドレス、および相手プログラマブルコントローラからの共有接点情報の格納エリアの開始チャネルが格納されている。

【0049】従って、この通信ポート情報テーブルT2に示すように、共有接点読出しコマンドC1（図7参照）および後述する構成の共有接点読出しレスポンスR1（図10参照）にも含まれている通信ポート番号と、共有接点読出しレスポンスを受信した際に起動する共有接点ユーザプログラムの先頭アドレスとが対応してあり、かつ、各ユーザプログラムの先頭アドレスがメモリマップM（図2参照）により対応しているため、この通信ポート番号が、共有接点情報を使用する共有接点ユーザプログラムを指示するプログラム指示情報ということになる。

【0050】なお、本実施例では、このように通信ポート番号をプログラム指示情報として使用しているが、共有接点ユーザプログラムのプログラム番号をプログラム指示情報として直接使用して、共有接点読出しコマンドC1や共有接点読出しレスポンスR1に通信ポート番号と共に、あるいは通信ポート番号の代わりに共有接点ユーザプログラムのプログラム番号をセットするようにしても良い。

【0051】次に、共有接点読出しコマンドを受信した相手プログラマブルコントローラ側における共有接点情報の送信処理等を説明する。

【0052】図9に、共有接点読出しコマンドを受信した相手プログラマブルコントローラ側における共有接点情報の送信処理を示す。

【0053】共有接点読出しコマンドを受けた相手プログラマブルコントローラでは、まず、その共有接点読出しコマンドから当該要求元プログラマブルコントローラのネットワークアドレスを取り出して、共有接点読出しレスポンスの相手プログラマブルコントローラのネットワークアドレスにセットすると共に（ステップ700）、続いてその自プログラマブルコントローラのネッ

トワークアドレスを共有接点読出しレスポンスにセットし(ステップ710)、レスポンスコードをセットする(ステップ720)。

【0054】次いで、その共有接点読出しコマンド内の通信ポート番号を、そのまま共有接点読出しレスポンスにセットし(ステップ740)、さらに受信側の相手プログラマブルコントローラの接点情報開始チャネルと、受信要求チャネル数とを取り出し、そのチャネルより要求チャネル数分の接点情報を読出して、その読出したチャネル数と共に共有接点読出しレスポンスにセットして(ステップ750)、その共有接点読出しレスポンスを送信し(ステップ760)、この処理を終了する。

【0055】図10に、図9の処理により共有接点読出しコマンドを送信してきたプログラマブルコントローラに返送される共有接点読出しレスポンスのフォーマットを示す。

【0056】共有接点読出しレスポンスR1は、図に示すように、共有接点読出しコマンドの送信元である相手プログラマブルコントローラのネットワークアドレス、共有接点読出しコマンドの受信側である自プログラマブルコントローラのネットワークアドレス、コマンド/レスポンスコード、通信ポート番号、読出した共有接点情報のチャネル数、および読出した共有接点情報から構成されて送信される。

【0057】次に、図4におけるステップ600の共有接点処理を詳細に説明する。

【0058】図11に、図4におけるステップ600の共有接点処理を詳細に示す。

【0059】この図11に示す共有接点処理は、独自接点処理中、すなわち独自接点ユーザプログラム中のN\_READ命令の実行により共有接点読出しコマンドを送信したプログラマブルコントローラが、その共有接点読出しレスポンスを受信した場合の処理を示している。

【0060】具体的には、MPU11は、相手プログラマブルコントローラからネットワークNを介して共有接点読出しレスポンスを受信すると、まずは、図8に示す通信ポート情報テーブルT2から通信ポート番号に対応する相手プログラマブルコントローラの共有接点情報格納エリアの開始チャネルを検索して、そのチャネルを介し受信した共有接点読出しレスポンスに含まれている共有接点情報を読出して(ステップ610)、さらに独自接点ユーザプログラムの実行中か否かを判断する(ステップ620)。

【0061】ここで、独自接点ユーザプログラムの実行中である場合には(ステップ620“Yes”)、その独自接点ユーザプログラムの実行を中断して(ステップ630)、さらにその中断した独自接点ユーザプログラムの実行情報をワークRAM13等の任意の退避エリアに退避させる(ステップ640)。

【0062】そして、通信ポート情報テーブルT2より、通信ポート番号に対応する共有接点ユーザプログラムの先頭アドレスを取り出し、そのプログラム中にEND命令を発見するまで上記送信してきた共有接点情報を使用してその共有接点ユーザプログラムを実行し続け(ステップ650)、その共有接点ユーザプログラムの実行終了後、割込により独自接点ユーザプログラム等の実行を中断していたか否かを判断し(ステップ660)、独自接点ユーザプログラムの実行を中断していた場合のみ(ステップ660“Yes”)、ワークRAM13等に退避させていたその実行情報を復帰させ(ステップ670)、中断していた独自接点ユーザプログラムの実行を再開させて(ステップ680)、以上の共有接点処理を終了する。

【0063】尚、共有接点ユーザプログラムの実行終了後、実行を中断していた他の共有接点ユーザプログラムがある場合、すなわち異なる共有接点ユーザプログラムに共有接点情報が連続して入力していた場合には、中断していた他の共有接点ユーザプログラムも同様に実行を再開させ、その後実行を中断していた独自接点ユーザプログラムの実行を再開させるようにする。

【0064】次に、本実施例における独自接点ユーザプログラムおよび共有接点ユーザプログラムの具体例を説明する。

【0065】図12に、上記第1実施例のプログラマブルコントローラにより構成されたPCシステム全体を示す。

【0066】このPCシステムでは、図に示すように、本実施例に係る3台のプログラマブルコントローラPC(A)～(C)がネットワークNを介して構成されており、各々が、図1に示すような構成を有している。

【0067】このシステムでは、ネットワークNを介した共有接点情報等のデータの読出しは、上述の通りチャネル単位で行われているため、PC(A)は、接点A1～A4のチャネルA、PC(B)は、接点B1、B2のチャネルB、PC(C)は、接点C1、C2のチャネルCとして指定するものとする。なお、PC(A)～(C)は、内部接点エリアとしてチャネルX、Yを有しており、通常接点としてチャネルX1、X2、・・・や、Y1、Y2、・・・をユーザプログラムの実行に利用できるように構成されている。

【0068】ここでは、PC(A)は、入力接点A1がONになると、PC(B)の入力接点B1、B2をネットワークNを介して読出し、入力接点B1、B2の双方がONの場合にその出力接点A3をONにし、入力接点A2がONになると、PC(C)の入力接点C1、C2をネットワークNを介して読出し、入力接点C1、C2の双方がONの場合にその出力接点A4をONにする、というアプリケーション処理を想定している。

【0069】図13(a)～(c)に、上記アプリケーション処理を想定した場合のPLC(A)におけるラダ



一言語で記述したユーザプログラムの具体例を示す。

【0070】(a)は、図1に示す本実施例の構成では、ユーザプログラムメモリ15aに格納される独自接点ユーザプログラムを示している。

【0071】具体的には、この独自接点ユーザプログラムP1は、接点A1がONになると、通信ポート0番を使用して、PC(B)のチャネルBより共有接点情報を読出し、その共有接点情報を含むレスポンスが返送されると、チャネルXに書き込んだ後、ユーザプログラムメモリ15bに格納される共有接点ユーザプログラムP2を讀出して起動する。次いで、接点A2がONになると、通信ポート1番を使用して、PC(C)のチャネルCより共有接点情報を讀出し、その共有接点情報を含むレスポンスが返送されると、チャネルYに書き込んだ後、ユーザプログラムメモリ15cに格納された他の共有接点ユーザプログラムP3を讀出し起動する、というプログラムを示している。

【0072】(b)は、ユーザプログラムメモリ15bに格納される共有接点ユーザプログラムP2を示しており、具体的には、接点X1とX2とが共にONのとき、接点A3をONにする、というプログラムを示している。

【0073】(c)は、ユーザプログラムメモリ15cに格納される共有接点ユーザプログラムP3を示しており、具体的には、接点Y1とY2とが共にONのとき、接点A4をONにする、というプログラムを示している。

【0074】よって、以上説明した本実施例における命令実行やコマンド/レスポンス処理を行うことにより、この図13(a)～(c)に示すユーザプログラムを処理するだけで、独自接点ユーザプログラムP1中の第1のN\_READ命令を実行することによって、ネットワークNを介した共有接点情報を受信した後、その共有接点情報を使用して共有接点ユーザプログラムP2が実行される一方、独自接点ユーザプログラムP1中の第2のN\_READ命令を実行することによって、ネットワークNを介した共有接点情報を受信した後、その共有接点情報を使用して共有接点ユーザプログラムP2が実行されることになる。

【0075】従って、本実施例によれば、ユーザプログラムを共有接点ユーザプログラムと、独自接点ユーザプログラムとに分けて格納し、通常はネットワークNを介した通信と関係のない独自接点ユーザプログラムのみが実行される一方、共有接点情報がネットワークを介して送信されてきた場合には、割込みによってその共有接点情報を使用して共有接点ユーザプログラムが実行されるので、共有接点ユーザプログラムと独自接点ユーザプログラムとが混在して1つになっていた従来のプログラマブルコントローラと比較して、ユーザプログラムの実行が接点情報の変化に対しリアルタイムで追従することが

可能になる。

【0076】また、ネットワークを介し他のプログラマブルコントローラから共有接点情報が送られてきない場合には、共有接点ユーザプログラムを実行しないため、このような場合でも共有接点ユーザプログラムと独自接点ユーザプログラムとが混在して1つになっている共有接点ユーザプログラムを実行してしまう従来のプログラマブルコントローラと比較して、サイクルタイムを短縮化できることになる。

【0077】さらに、本実施例では、プログラマブルコントローラAが複数の共有接点ユーザプログラムを有し、通信ポートに対応させて各共有接点ユーザプログラムを実行するようにしており、かつ、相手プログラマブルコントローラから送信されてくる共有接点読出しレスポンスには、共有接点情報だけでなく、その共有接点情報を使用する共有接点ユーザプログラムに対応させた通信ポート番号を含めるようにしたので、共有接点読出しレスポンスを受信したプログラマブルコントローラでは、その共有接点読出しレスポンスに含まれている共有接点情報を使用して、複数の共有接点ユーザプログラムの中からその通信ポート番号に対応した共有接点ユーザプログラムを選択して実行でき、共有接点処理の応用性および弾力性が向上する。

【0078】次に、本発明に係るプログラマブルコントローラの第2実施例を説明する。

【0079】この第2実施例のプログラマブルコントローラでは、ハード構成等は図1に示す上記第1実施例とは同じだが、この第2実施例では、上記第1実施例とは異なり、共有接点ユーザプログラムを実行するプログラマブルコントローラ側で共有接点ユーザプログラムを所有する必要がなく、図1に示す第1実施例の構成において、独自接点ユーザプログラムを格納したユーザプログラムメモリ15aのみを有し、共有接点ユーザプログラムを格納したユーザプログラムメモリ15b、15cを有していないことを特徴としている。

【0080】つまり、この第2実施例のプログラマブルコントローラは、上記第1実施例とは異なり、N\_READ命令の実行により他のプログラマブルコントローラへ共有接点読出しコマンドの送信、および当該プログラマブルコントローラから共有接点読出しレスポンスの受信により共有接点ユーザプログラムを実行するのではなく、他のプログラマブルコントローラによるN\_WRITE命令の実行により、ネットワークを介した当該他のプログラマブルコントローラからの共有接点情報、および当該共有接点情報を使用して実行すべき共有接点ユーザプログラムを含む共有接点書き込みコマンドの受信により、共有接点ユーザプログラムの実行処理を行うことを特徴としている。

【0081】従って、N\_WRITE命令は、上述のようにユーザプログラム中に記述されるが、上記第1実施

例と同様に、その命令と共に、この命令の送信となる相手プログラマブルコントローラのネットワークアドレスや、通信ポート番号（複数のプログラマブルコントローラに同時に共有接点読出しコマンドを送信する場合には、そのレスポンスを区別するための識別子となる。）、この命令に基づく共有接点書き込みコマンドを受信した相手プログラマブルコントローラ側で実行させる共有接点ユーザプログラムのプログラム番号、書き込むべき自プログラマブルコントローラにおける共有接点情報の開始チャネル、相手プログラマブルコントローラにおける自プログラマブルコントローラからの共有接点情報の格納エリア開始チャネル、および書き込みを要求するチャネル数、等のパラメータがユーザプログラム中に記述されることとなる。

【0082】図14に、この第2実施例における共有接点書き込みコマンドの送信処理を詳細に示す。

【0083】この共有接点書き込みコマンドを送信するプログラマブルコントローラ側で、MPU等が独自接点ユーザプログラム中のN\_WRITE命令を実行すると、まずは、そのN\_WRITE命令の上記パラメータから相手プログラマブルコントローラのネットワークアドレスを取り出し、送信すべき共有接点書き込みコマンドにセットすると共に（ステップ800）、自プログラマブルコントローラのネットワークアドレスも共有接点書き込みコマンドにセットして（ステップ810）、さらにコマンドコードをセットする（ステップ820）。

【0084】次に、そのパラメータから通信ポート番号を取り出し、共有接点書き込みコマンドにセットすると共に（ステップ830）、そのパラメータから受信しようとする自プログラマブルコントローラにおける共有接点情報格納エリアの開始チャネルを取り出して、その共有接点書き込みコマンドにセットし（ステップ840）、さらに書き込み要求チャネル数も共有接点書き込みコマンドにセットする（ステップ850）。

【0085】次いで、さらにN\_WRITE命令のパラメータから、書込もうとする自プログラマブルコントローラの共有接点情報の開始チャネルと、書き込み要求チャネル数とを取り出して、そのチャネルより指定チャネル数分の共有接点情報を讀出して、書き込み要求チャネル数と共に共有接点書き込みコマンドにセットし（ステップ860）、さらにそのパラメータからユーザプログラム番号を取り出し、メモリプログラムMやユーザプログラムアドレス管理テーブルT1を参照してそのユーザプログラムの先頭アドレスを検索し、その先頭アドレスのユーザプログラムの共有接点書き込みコマンドにコピーして（ステップ870）、指定ポートを使用して以上のように設定した共有接点書き込みコマンドを通信相手プログラマブルコントローラへ送信する（ステップ880）。

【0086】図15は、図14におけるステップ880の処理で送信される共有接点書き込みコマンドのフォーマットを示している。

【0087】この共有接点書き込みコマンドC2は、上記図14に示す処理によって、相手プログラマブルコントローラのネットワークアドレス、自プログラマブルコントローラのネットワークアドレス、コマンド/レスポンスコード、通信ポート番号、自プログラマブルコントローラの共有接点情報を格納するエリアの開始チャネル、書き込みを要求するチャネル数、書き込みを要求する共有接点情報、およびその共有接点情報書き込みを使用して実行される共有接点ユーザプログラム、から構成されることになる。

【0088】次に、共有接点書き込みコマンドの受信側のプログラマブルコントローラの処理について説明する。

【0089】図16に、共有接点書き込みコマンドの受信側のプログラマブルコントローラの処理を詳細に示す。

【0090】まず、図15に示すように構成された共有接点書き込みコマンドを受信したプログラマブルコントローラでは、その共有接点書き込みコマンドから、書き込み要求チャネル数分の接点情報を讀出し、要求された格納エリア開始チャネルより書き込み（ステップ900）、次いで独自接点等のユーザプログラムの実行中か否かを判断する（ステップ910）。

【0091】ここで、独自接点等のユーザプログラムの実行中であると判断した場合には（ステップ910“YES”）、そのユーザプログラムの実行を中断し（ステップ920）、さらにその中断したユーザプログラムの実行情報をワークRAM13等の任意のエリアに退避する（ステップ930）。

【0092】そして、そのネットワークを介して受信した共有接点書き込みコマンド内に格納されている共有接点情報と、共有接点ユーザプログラムとを讀出してワークRAM13等に格納し、その讀出した共有接点情報を用いて同様に讀出した共有接点ユーザプログラムを、そのユーザプログラム中にEND命令を発見するまで実行し続ける（ステップ940）。

【0093】共有接点ユーザプログラムの実行終了後は、共有接点書き込みコマンドに対するレスポンスを要求元に返送し（ステップ950）、続いてユーザプログラムの実行を中断していたか否かを判断して（ステップ960）、独自接点等のユーザプログラムの実行を中断していた場合のみ（ステップ960“YES”）、ワークRAM13等に退避させていたその実行情報を復帰させ（ステップ970）、その実行情報を復帰した後、中断していた独自接点等のユーザプログラムの実行を再開させて（ステップ980）、以上の処理を終了する。

【0094】尚、共有接点ユーザプログラムの実行終了後、実行を中断していた他の共有接点ユーザプログラムがある場合には、その他の共有接点ユーザプログラムも同様にして実行を再開させ、その後に実行を中断していた独自接点ユーザプログラムの実行を再開させるように

する。

【0095】図17に、図16におけるステップ950の処理で、共有接点書込みコマンドを送信してきたプログラマブルコントローラに返送する共有接点書込みレスポンスのフォーマットを示す。

【0096】この共有接点書込みレスポンスR2は、図に示すように、相手プログラマブルコントローラのネットワークアドレスと、自プログラマブルコントローラのネットワークアドレスと、コマンド/レスポンスコードと、通信ポート番号とにより構成されており、これが共有接点書込みコマンドを送信してきたプログラマブルコントローラに返送されることになる。

【0097】従って、この第2実施例によれば、共有接点情報と共に、その共有接点情報を使用する共有接点ユーザプログラムを含む共有接点書込みコマンドを、ネットワークを介してプログラマブルコントローラ間で通信するようにし、その共有接点書込みコマンドを受信したプログラマブルコントローラでは、その送信されてきた共有接点情報を使用して、同様に送信されてきた共有接点ユーザプログラムを実行するようにしたため、共有接点ユーザプログラムを実行するプログラマブルコントローラ側では、その共有接点ユーザプログラムを所有する必要がなくなり、共有接点ユーザプログラムを格納するメモリを設ける必要がなくなると共に、共有接点ユーザプログラムの共用化および分散化が可能になる。

【0098】このため、この第2実施例によれば、上記第1実施例と同様に、共有接点ユーザプログラムと独自接点ユーザプログラムとが混在して1つになっていた従来のプログラマブルコントローラと比較して、接点情報の変化にリアルタイムで追従することが可能になると共に、ネットワークを介して他のプログラマブルコントローラから共有接点情報やその共有接点情報を使用する共有接点ユーザプログラムが送信されてこない場合には、共有接点ユーザプログラムを実行する必要がないので、サイクルタイムを短縮化できることになる。

【0099】

【発明の効果】以上説明したように、本発明では、ユーザプログラムを共有接点ユーザプログラムと、独自接点ユーザプログラムとに分けて格納しておき、通常はネットワークを介した通信と関係のない独自接点ユーザプログラムのみを実行する一方、共有接点情報がネットワークを介して送信されてきた場合には、割込みによってその共有接点情報を使用して共有接点ユーザプログラムを実行するようにしたため、共有接点ユーザプログラムと独自接点ユーザプログラムとが混在して1つになっていた従来のプログラマブルコントローラと比較して、ユーザプログラムの実行が接点情報の変化にリアルタイムで追従することが可能になる。

【0100】また、ネットワークを介して他のプログラマブルコントローラから共有接点情報が入力しない場合に

は、共有接点ユーザプログラムを実行しないため、このような場合でも共有接点ユーザプログラムと独自接点ユーザプログラムとが混在して1つになっているため共有接点ユーザプログラムを実行してしまう従来のプログラマブルコントローラと比較して、サイクルタイムを短縮化できることになる。

【0101】さらに、本発明では、ネットワークを介して共有接点情報だけでなくその共有接点情報を使用する共有接点ユーザプログラムを指示するプログラム指示情報も送信するようにしているため、共有接点情報等を受信したプログラマブルコントローラでは、その共有接点情報を使用して複数の共有接点ユーザプログラムの中から対応した共有接点ユーザプログラムを選択して実行でき、共有接点ユーザプログラムの実行の応用性および弾力性が向上する。

【0102】またさらに、本発明では、共有接点情報と共に、その共有接点情報を使用する共有接点ユーザプログラムをネットワークを介してプログラマブルコントローラ間で通信するようにし、その共有接点情報および共有接点ユーザプログラムを受信したプログラマブルコントローラにその共有接点情報を使用して、同様に送信されてきた共有接点ユーザプログラムを実行させるようにもしているため、従来のプログラマブルコントローラと比較して、上述のように接点情報の変化にリアルタイムで追従できると共に、サイクルタイムを短縮化できるだけでなく、共有接点ユーザプログラムを実行するプログラマブルコントローラ側ではその共有接点ユーザプログラムを所有する必要がなくなることにより、共有接点ユーザプログラムを格納するメモリを設ける必要がなくなると共に、共有接点ユーザプログラムの共用化および分散化が可能になる。

【図面の簡単な説明】

【図1】本発明に係るプログラマブルコントローラの実施例の構成を示す説明図。

【図2】ユーザプログラムメモリ15a~15cに格納されるユーザプログラムのメモリマップを示す説明図。

【図3】ユーザプログラム管理テーブルを示す説明図。

【図4】プログラマブルコントローラの全体の処理を示すフローチャート。

【図5】図4におけるステップ200の通常接点処理を詳細に示すフローチャート。

【図6】図5におけるステップ230共有接点読出しコマンドの送信処理を詳細に示すフローチャート。

【図7】図6におけるステップ237の処理で送信される共有接点読出しコマンドのフォーマットを示す説明図。

【図8】図6に示すステップ238等の処理で作成される通信ポート情報テーブルT2を示す説明図。

【図9】共有接点読出しコマンドを受信した相手プログラマブルコントローラ側における共有接点情報の送信処

理を示すフローチャート。

【図10】図9の処理により共有接点読出しコマンドを送信してきたプログラマブルコントローラに返送するレスポンスメッセージのフォーマットを示す説明図。

【図11】図4におけるステップ400の共有接点処理を詳細に示すフローチャート。

【図12】第1実施例のプログラマブルコントローラにより構成されたPCシステム全体を示す説明図。

【図13】独自接点および共有接点のユーザプログラムの具体例を示す説明図。

【図14】共有接点書き込みコマンドの送信処理を詳細に示す説明図。

【図15】共有接点書き込みコマンドのフォーマットを示す説明図。

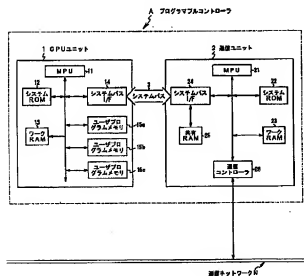
【図16】共有接点書き込みコマンドの受信側のプログラマブルコントローラの処理を詳細に示すフローチャート。

【図17】共有接点書き込みコマンドを送信してきたプログラマブルコントローラに返送するレスポンスメッセージのフォーマットを示す説明図。

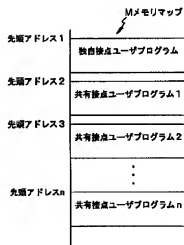
【符号の説明】

- A プログラマブルコントローラ (PC)  
B プログラマブルコントローラ  
C プログラマブルコントローラ  
N ネットワーク

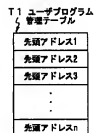
【図1】



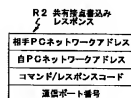
【図2】



【図3】

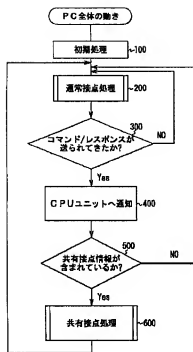


【図17】

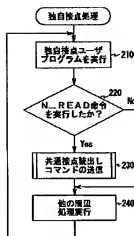


- 1 CPUユニット  
11 MPU (プログラム実行手段)  
12 システムROM  
13 ワークRAM  
14 システムバスインタフェース  
15a ユーザプログラムメモリ (通常接点プログラムメモリ)  
15b ユーザプログラムメモリ (共有接点プログラムメモリ)  
15c ユーザプログラムメモリ (共有接点プログラムメモリ)  
2 通信ユニット  
21 MPU  
22 システムROM  
23 ワークRAM  
24 システムバスインタフェース  
25 共有RAM  
26 通信コントローラ  
3 システムバス  
C1 共有接点読出しコマンド  
C2 共有接点書き込みコマンド  
R1 共有接点読出しレスポンス  
R2 共有接点書き込みレスポンス  
T1 ユーザプログラム管理テーブル  
T2 通信ポート情報テーブル

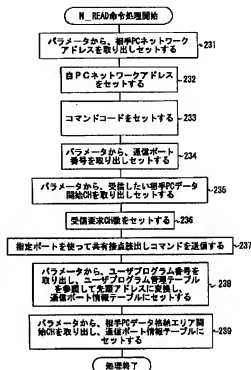
【図4】



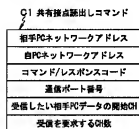
【図5】



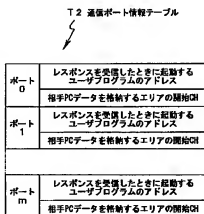
【図6】



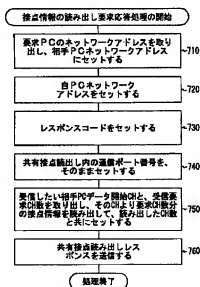
【図7】



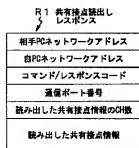
【図8】



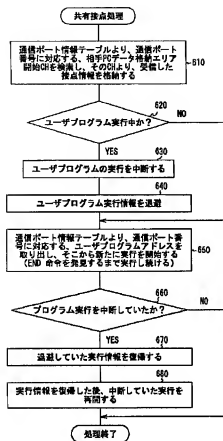
【図9】



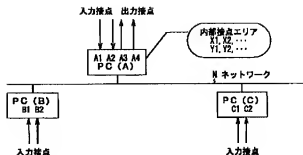
【図10】



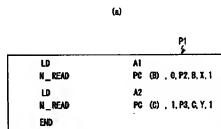
【図11】



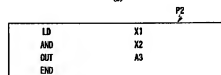
【図12】



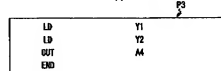
【図13】



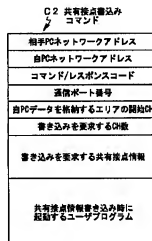
(b)



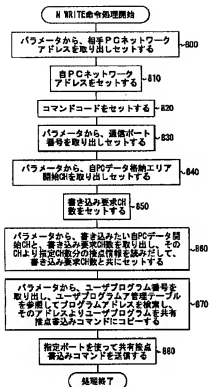
(c)



【図15】



【図14】



【図16】

